

Machine Learning 1

Lecture 9.2 - Unsupervised Learning
K-Means Clustering

Erik Bekkers

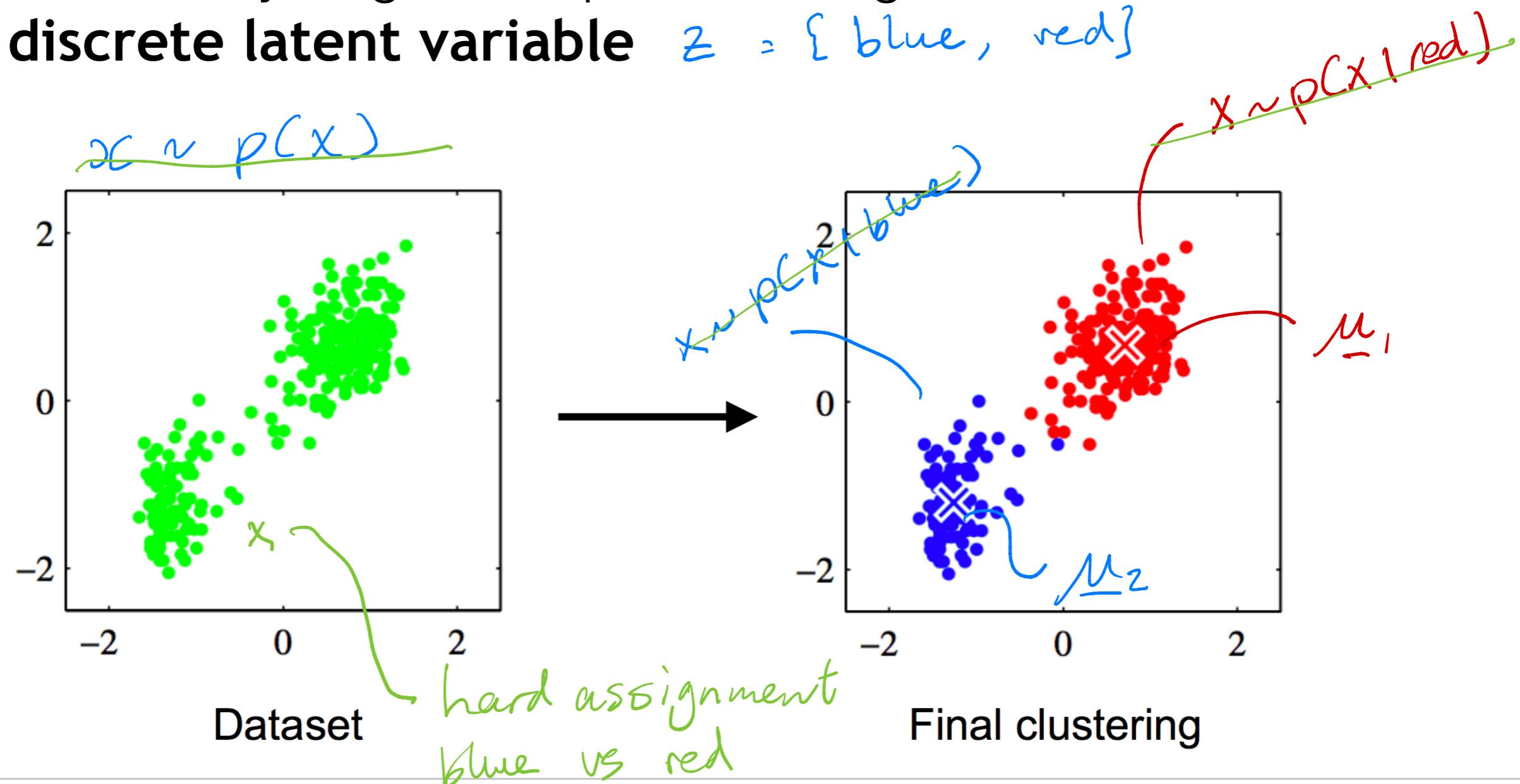
(Bishop 9.1)



Clustering with K-means

No probabilistic interpretation for now

- ▶ Data: a sample of points \mathbf{x} (without a target)
- ▶ Goal: every single data point is assigned to a cluster – a **discrete latent variable** $Z = \{\text{blue, red}\}$



K-means clustering as minimization problem

- Data: $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}, \mathbf{x}_n \in \mathbb{R}^D$

- Goal: partition into K clusters by minimizing $z_{nk} \in \{0, 1\}$

$$J = \sum_{n=1}^N \sum_{k=1}^K z_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

- Find cluster assignments (latent var) $z_{nk} \in \{0, 1\}$

and cluster means $\boldsymbol{\mu}_k \in \mathbb{R}^D$

- 1-hot-encoding: $z_{nk} = 1$ if and only if point n is assigned to cluster k

$$\underline{z}_h = \begin{pmatrix} z_{n1} \\ z_{n2} \\ \vdots \\ z_{nk} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ \vdots \end{pmatrix}$$

Minimize J (EM algorithm)

- ▶ Initialize with a random $\mu_k \in \mathbb{R}^D$

- ▶ Repeat until convergence:

- ▶ Find the **assignment** (fixed means) – **E-step**

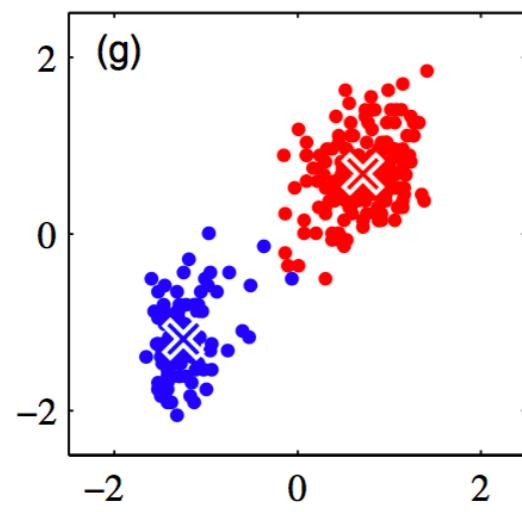
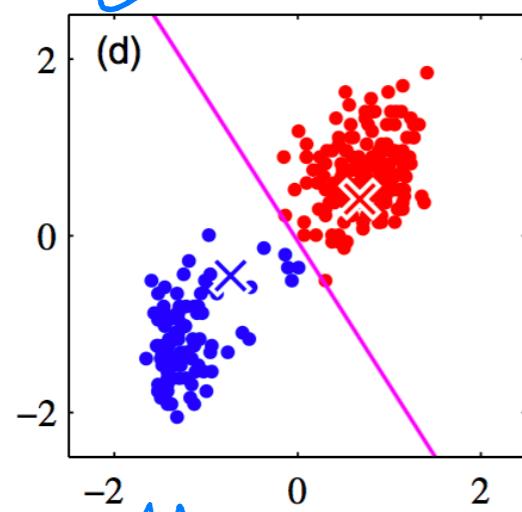
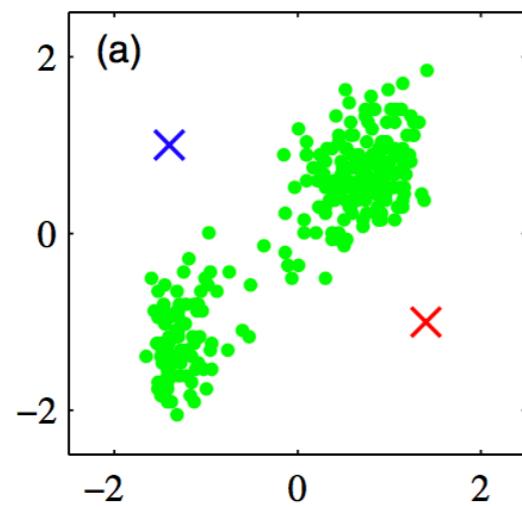
$$z_{nk} = \begin{cases} 1 & \text{if } k = \operatorname{argmin}_j \|x_n - \mu_j\|^2 \\ 0 & \text{otherwise} \end{cases}$$

- ▶ Find the means (fixed assignments) – **M-step**

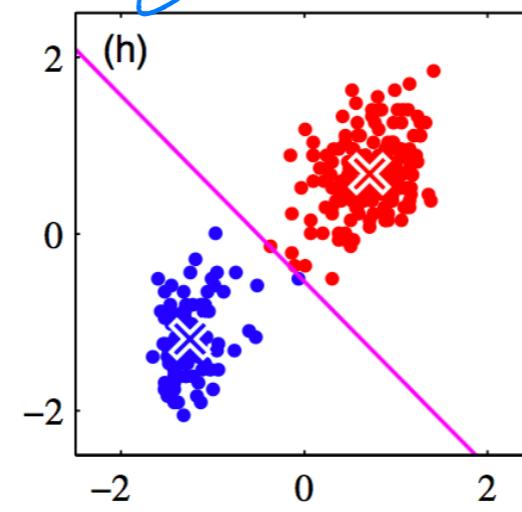
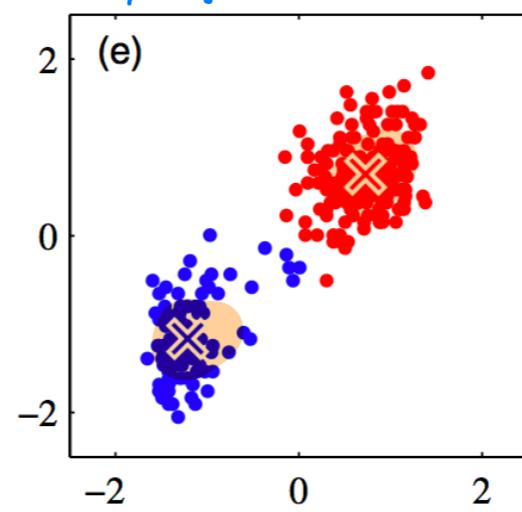
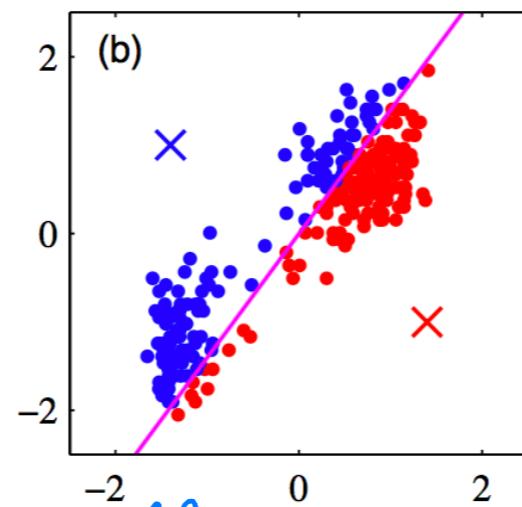
$$\mu_k = \frac{\sum_n z_{nk} x_n}{\sum_n z_{nk}}$$

"Expectation"
"Maximization"

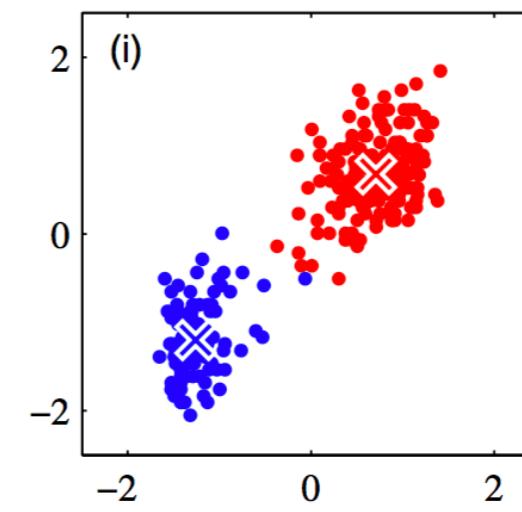
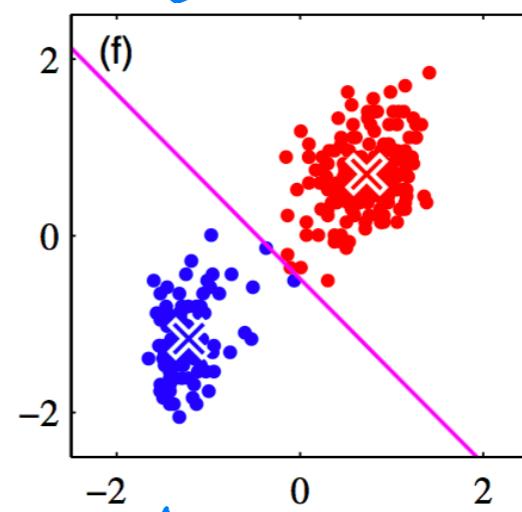
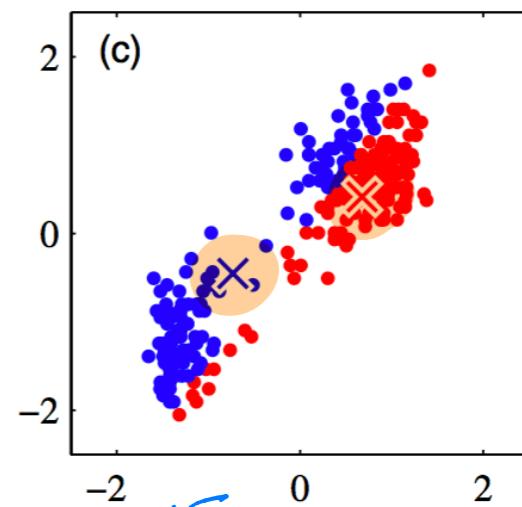
Example



(assignment)
E

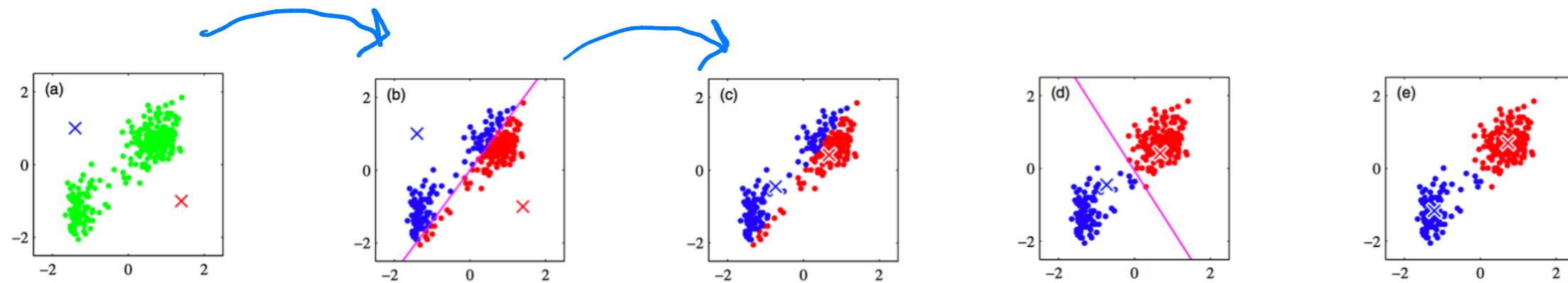
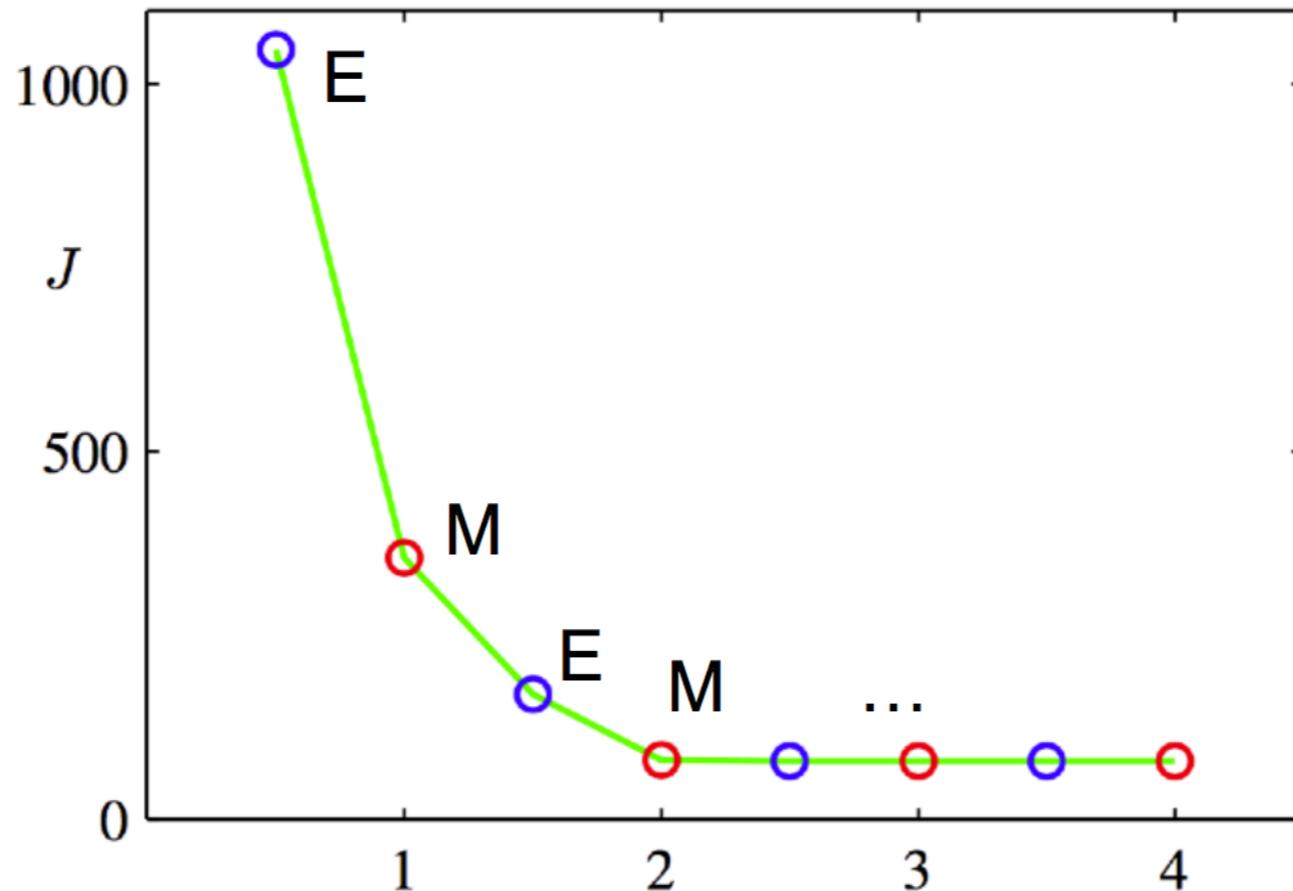


(update means)
M



Convergence

$$J = \sum_n \sum_k z_{nk} \|x_n - \mu_k\|^2$$



Initialize

E-step

M-step

E-step

M-step

But *global* convergence?

- ♦ J is non-convex for μ_k and z_{nk} together and k-means converges to a **local minimum**
- ♦ Can we do better? Random restarts with different initial cluster means and then select the clusters with minimal J .

Derivation of the M-step

$$J = \sum_{n=1}^N \sum_{k=1}^K z_{nk} \|x_n - \mu_k\|^2$$

- ♦ It is a convex function in μ_k (fixed z_{nk})

- ♦ Find the minimum by setting gradient = 0

$$\frac{\partial}{\partial \mu_k} \left(\sum_{n=1}^N \sum_{l=1}^K z_{nl} \|x_n - \mu_l\|^2 \right) = \sum_{n=1}^N z_{nk} \frac{\partial}{\partial \mu_k} \|x_n - \mu_k\|^2$$

≠ 0 only when l=k

$$= -2 \sum_{n=1}^N z_{nk} (x_n - \mu_k)^T = 0$$

- ♦ Hence

$$\sum_{n=1}^N z_{nk} x_n - \mu_k \sum_{n=1}^N z_{nk} = 0 \implies \mu_k = \frac{\sum_{n=1}^N z_{nk} x_n}{\sum_{n=1}^N z_{nk}}$$

Application: image compression

Data points : $x_n \in \{K, G, B\}$ pixel values
K clusters : color representation

1 1

2 2

1 1 1

X·Y integers
 $\in \{1, 2\}$

(g) (b)

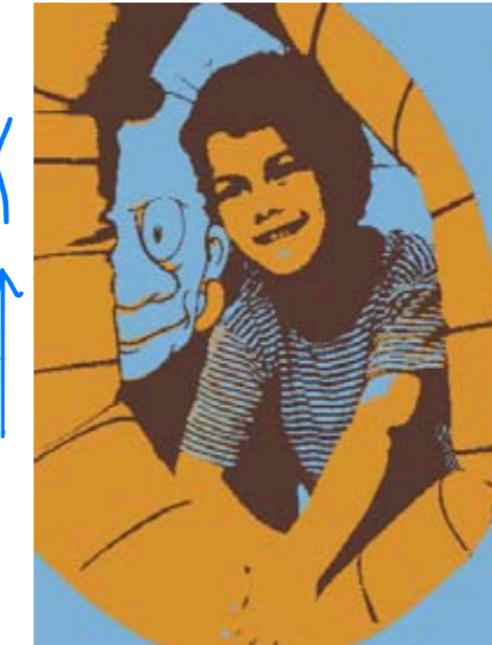
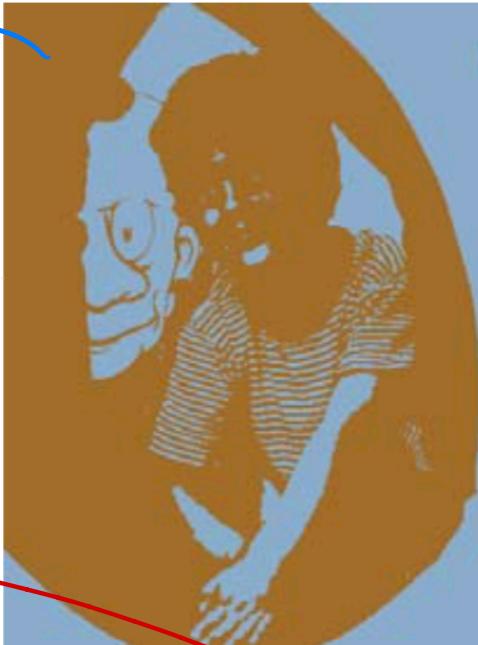
X·Y·3 values
 $\in [0, 255]$

$K = 2$

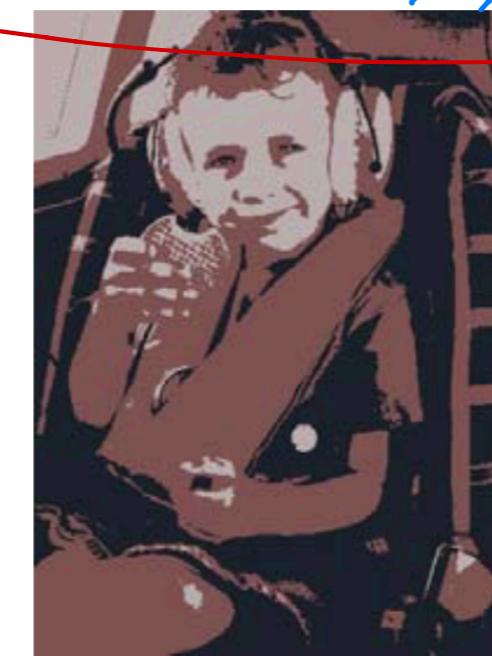
$K = 3$

$K = 10$

Original image

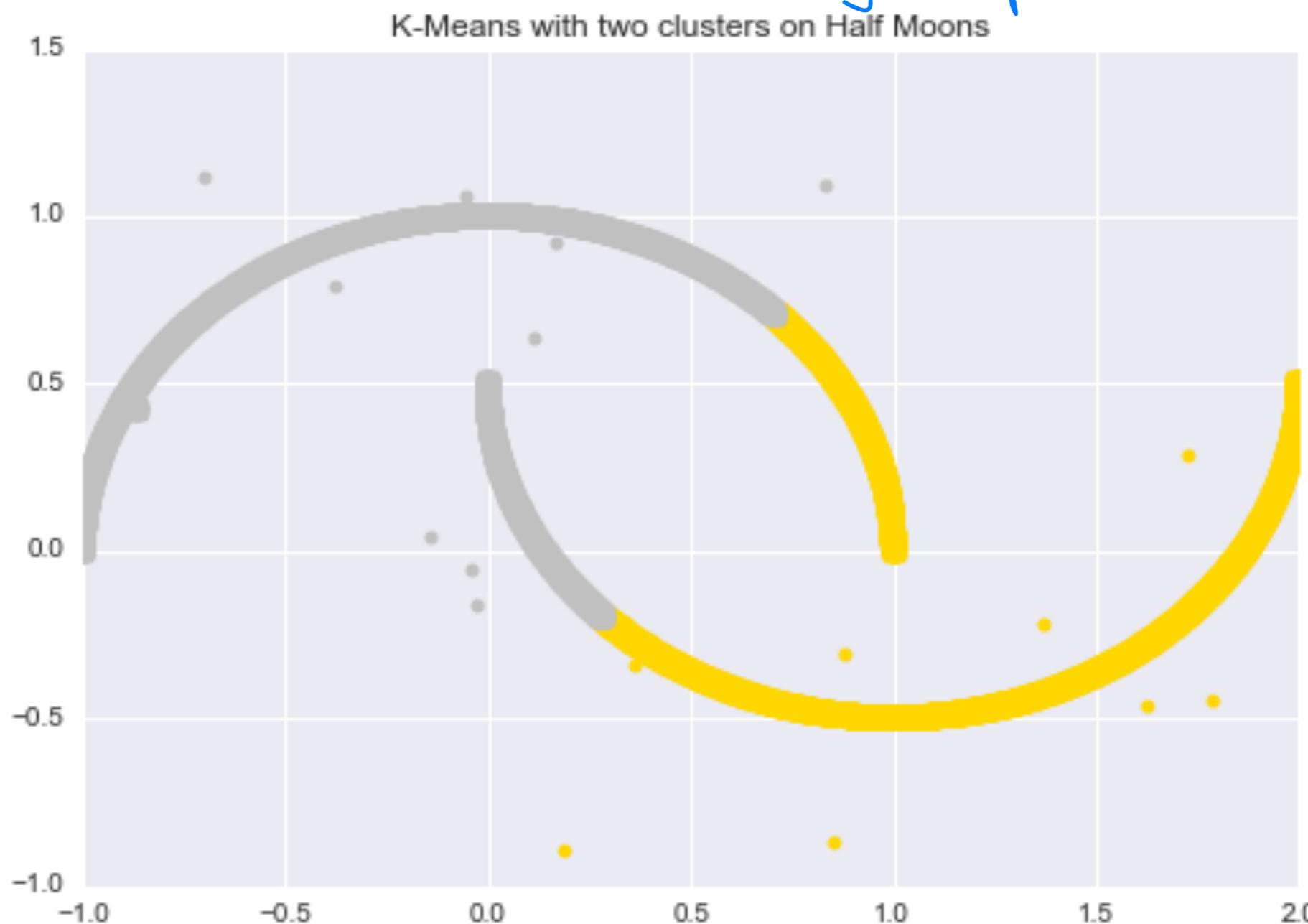


$\rightarrow X$



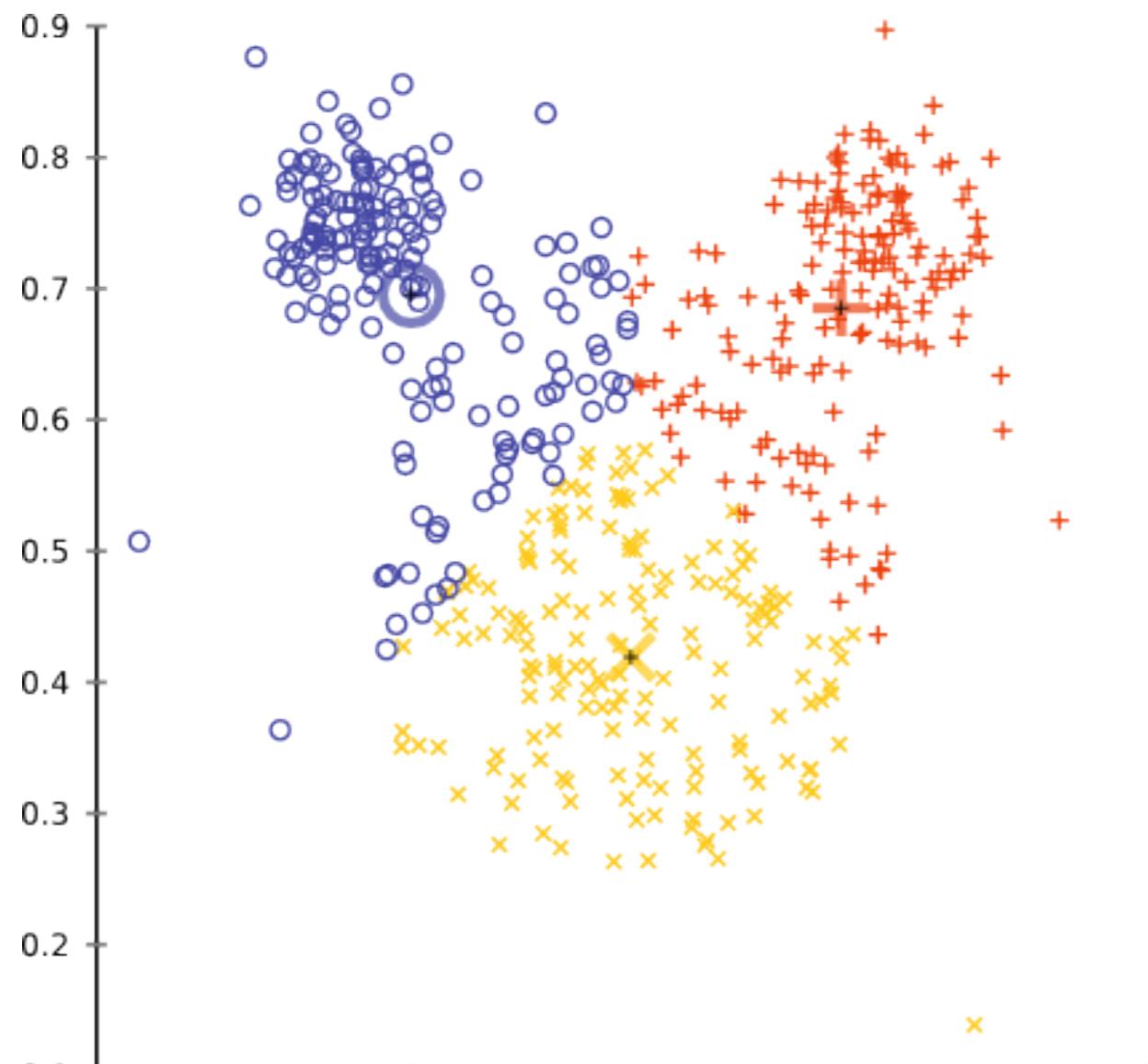
Failures of K-means

→ Only spherical clusters

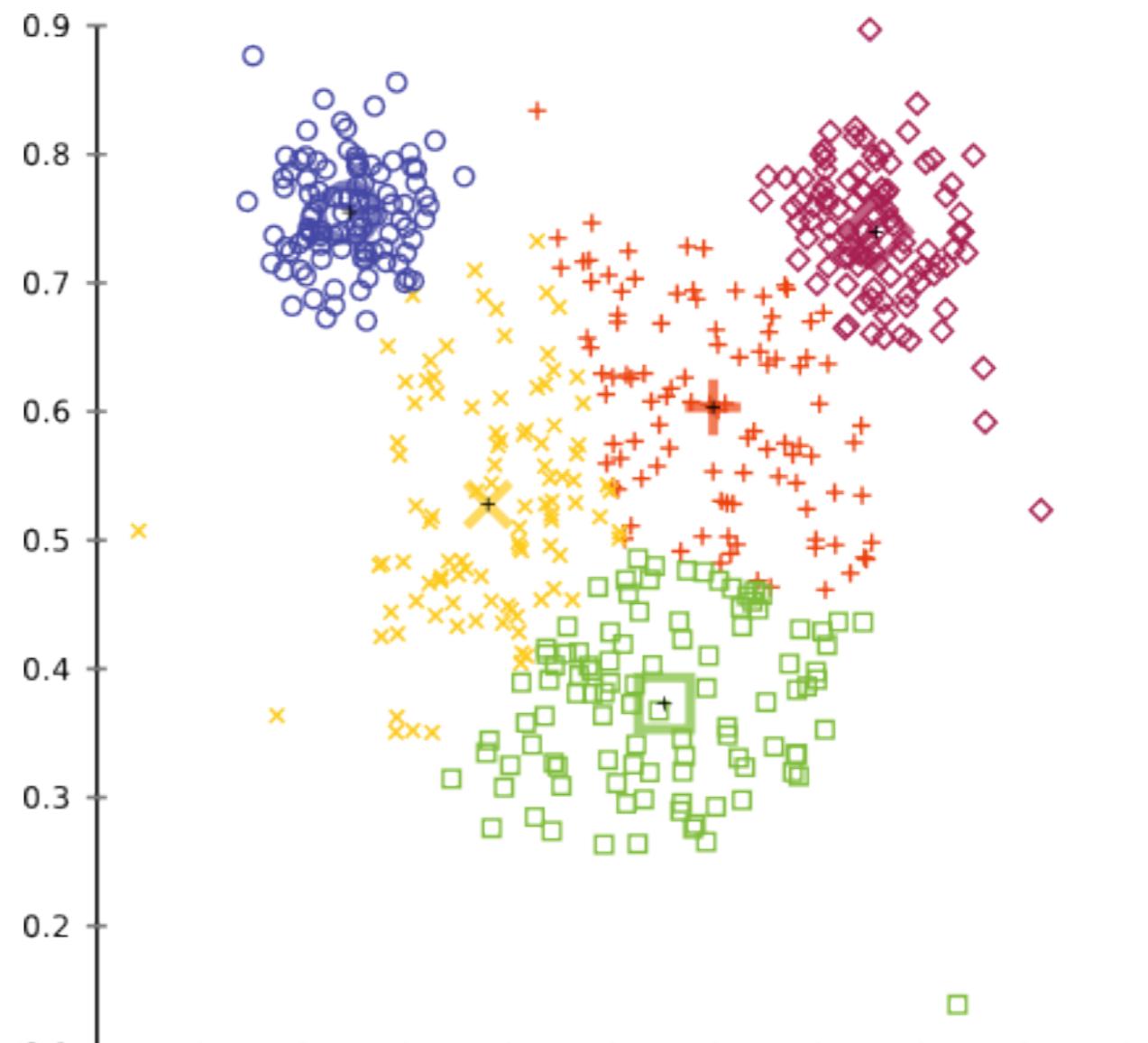


Failures of K-means (the mouse data!)

→ each cluster equal size



$K = 3$



$K = 5$

Improvements

$$J = \sum_{n=1}^N \sum_{k=1}^K z_{nk} \|x_n - \mu_k\|^2$$

- ▶ Stochastic gradient for big data
 - ▶ For each datapoint, update nearest cluster mean:

$$\begin{aligned}\mu_k &= \mu_k - \eta \left(\frac{\partial J}{\partial \mu_k} \right)^T \\ &= \mu_k + 2\eta(x_n - \mu_k)\end{aligned}$$

- ▶ Other distances between points (K-means)
 - ▶ Euclidean not always appropriate (discrete data), sensitive to outliers

$$\tilde{J} = \sum_{n=1}^N \sum_{k=1}^K z_{nk} \mathcal{V}(x_n, \underline{\mu}_k)$$

Pros & Cons

- ▶ Good
 - Simple to implement
 - Fast
- ▶ Bad
 - Local minima
 - Model only “spherical” clusters
 - Sensitive to the features scale
 - Number of clusters K to be chosen in advance
 - Cluster assignments are “hard”, not probabilistic => next topic, Gaussian Mixture Model

